myPass Report

Brukli Peshku
Hawraa Radhi
Ali Cheaitou

Project Overview

This project involves the development of a robust software system designed with key software engineering principles in mind, emphasizing maintainability and functionality through the implementation of design patterns. The project leverages the **Builder pattern** for constructing complex and secure password generation logic and the **Mediator pattern** to manage efficient interactions between UI components. These patterns ensured a modular and scalable system design.

Key Features

1. **Form Development**:
    - The project includes an "Add Identity" form, which allows users to input and manage data securely. Advanced validation techniques were integrated to ensure data accuracy and maintain the integrity of the system.
2. **Diagrams and Documentation**:
    - Comprehensive class diagrams were created to map the architecture and demonstrate how the Builder and Mediator patterns were implemented. These visual representations provided clarity for both developers and stakeholders.
3. **Design Pattern Integration**:
    - The **Builder pattern** was used to manage complex object creation, such as generating secure passwords with varying levels of complexity and additional features.
    - The **Mediator pattern** was implemented to streamline communication between UI components, reducing dependencies and improving system cohesion.

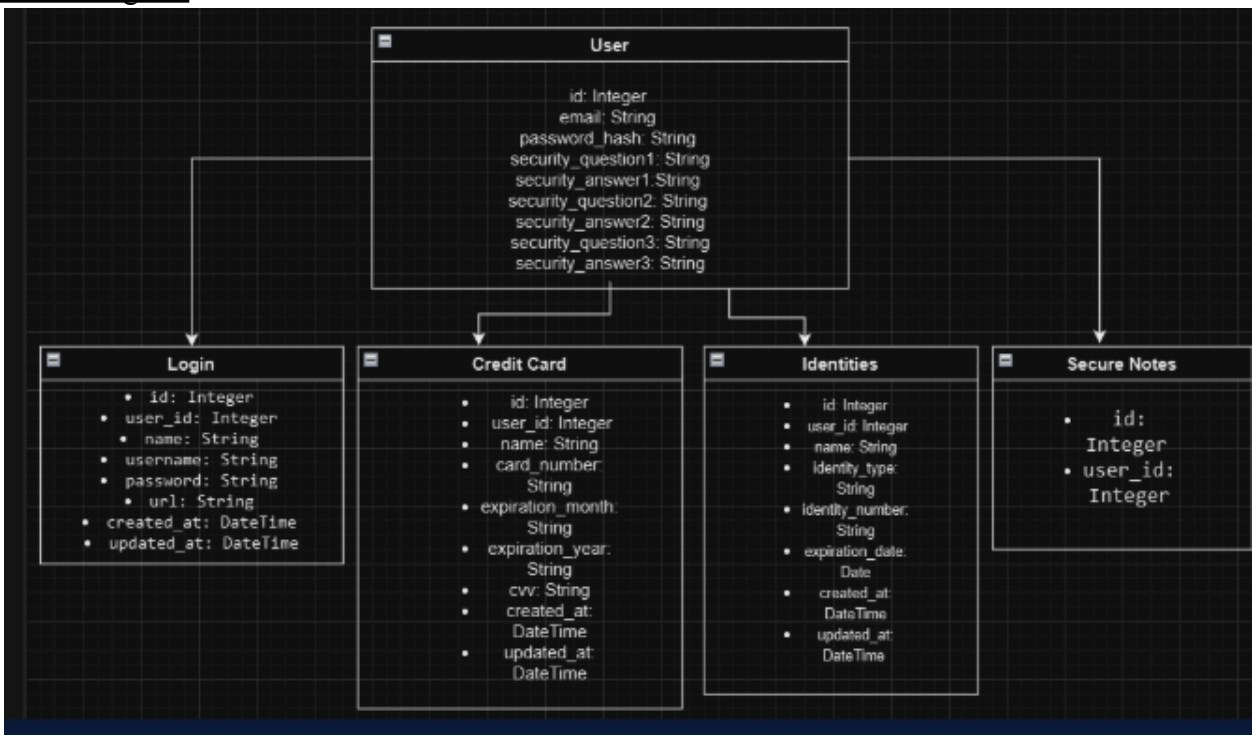
Database Used

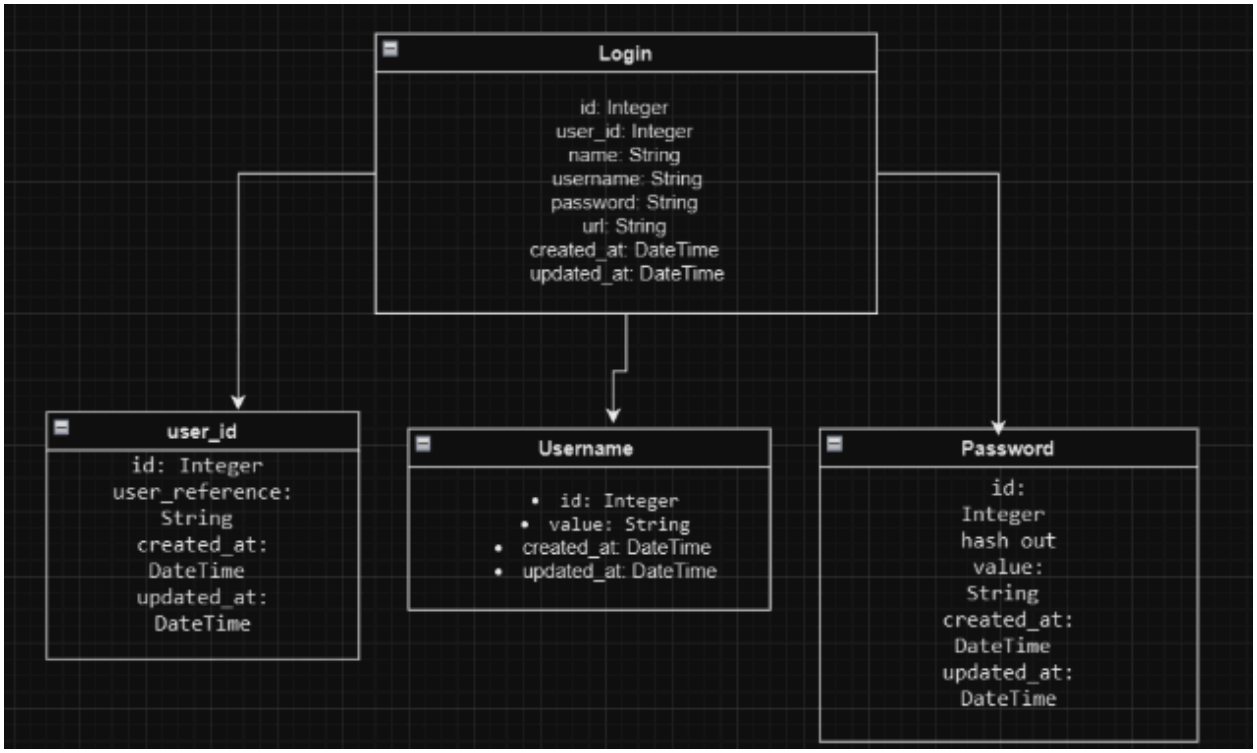
The project employed an **SQL database** to securely store user data, including identities and encrypted passwords. The database schema was carefully designed with normalization principles to ensure consistency and optimize performance. SQL queries were utilized to handle data retrieval, updates, and validations efficiently, supporting the system's advanced features.

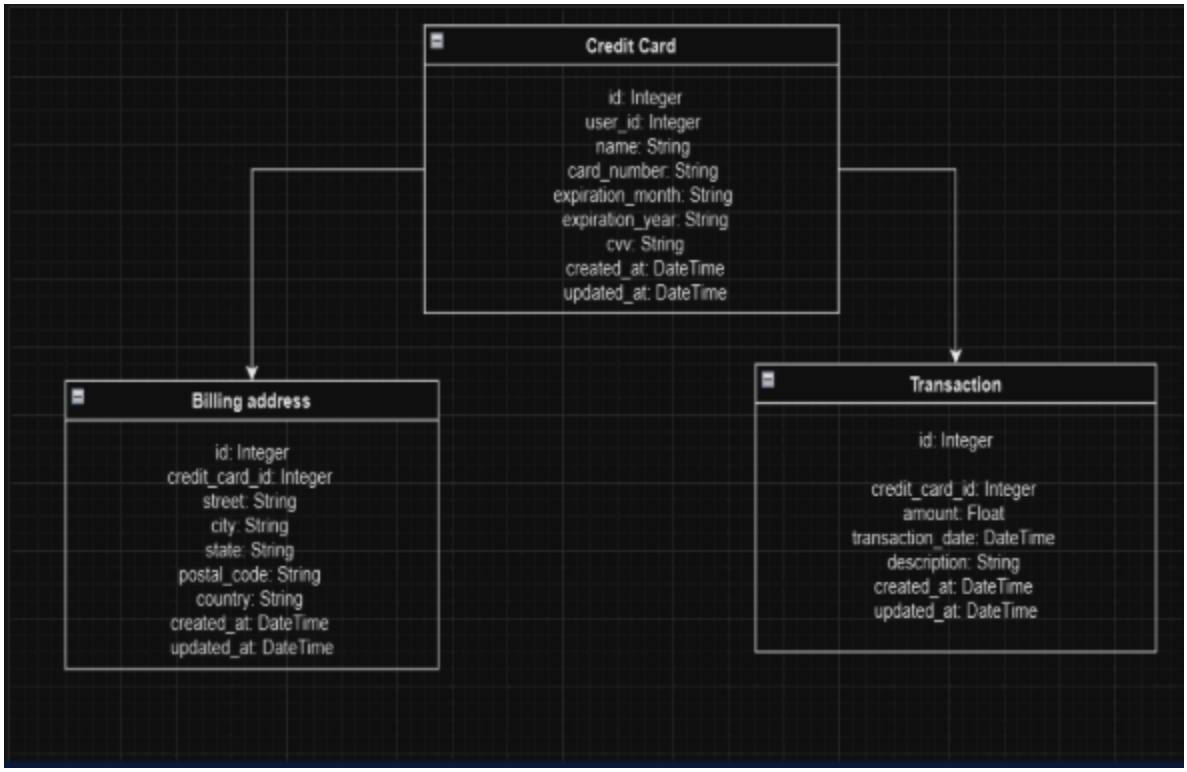
Personal Reflection

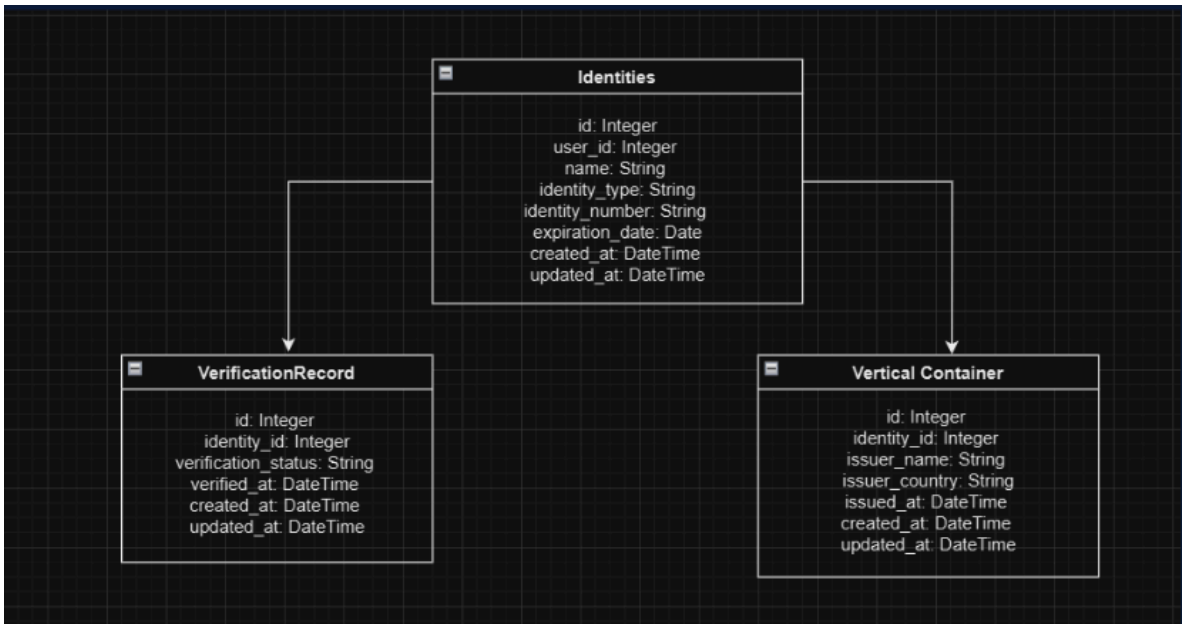
The most rewarding aspect of this project was integrating the **Mediator pattern** to simplify complex UI interactions. It was exciting to see how theoretical concepts like design patterns can address real-world software challenges. Additionally, working with an SQL database provided hands-on experience with managing and securing data in a professional and scalable manner.

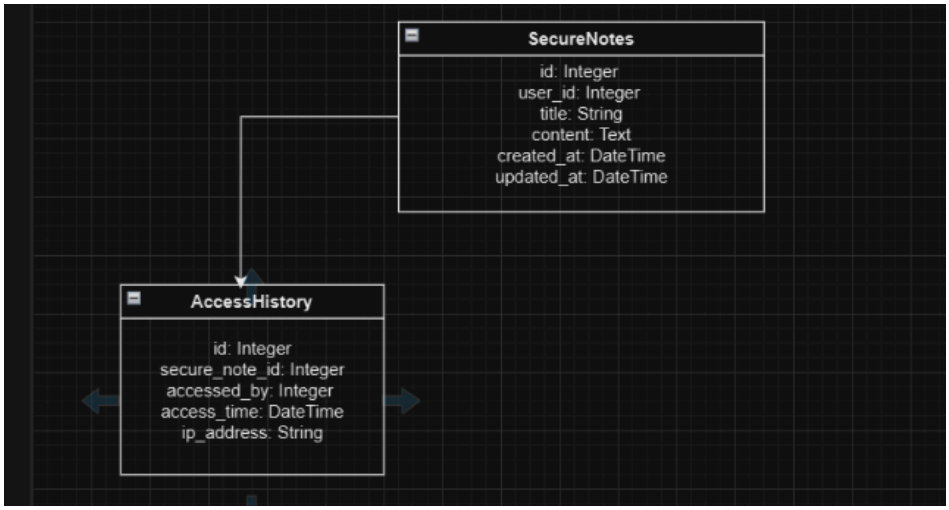
Class Diagram:






Credit Card
id: Integer
user_id: Integer
name: String
card_number: String
expiration_month: String
expiration_year: String
cvv: String
created_at: DateTime
updated_at: DateTime
Billing address
id: Integer
credit_card_id: Integer
street: String
city: String
state: String
postal_code: String
country: String
created_at: DateTime
updated_at: DateTime
Transaction
id: Integer
credit_card_id: Integer
amount: Float
transaction_date: DateTime
description: String
created_at: DateTime
updated_at: DateTime


Identities
id: Integer
user_id: Integer
name: String
identity_type: String
identity_number: String
expiration_date: Date
created_at: DateTime
updated_at: DateTime
VerificationRecord
id: Integer
identity_id: Integer
verification_status: String
verified_at: DateTime
created_at: DateTime
updated_at: DateTime
Vertical Container
id: Integer
identity_id: Integer
issuer_name: String
issuer_country: String
issued_at: DateTime
created_at: DateTime
updated_at: DateTime



Github Repository:

https://github.com/bruklipe/SoftwArchTermProj.git

Recoding:

https://youtu.be/x25PRUSF2Hg